

REMARKS

The above amendments to the above-captioned application along with the following remarks are being submitted as a full and complete response to the Office Action dated May 9, 2008. In view of the above amendments and the following remarks, the Examiner is respectfully requested to give due reconsideration to this application, to indicate the allowability of the claims, and to pass this case to issue.

Status of the Claims

As outlined above, claims 1-6 and 11 stand for consideration in this application, wherein claims 7-10 are being canceled without prejudice or disclaimer, while claims 1-3 are being amended. In addition, new claim 11 is hereby submitted for consideration.

All amendments to the application are fully supported therein. Applicants hereby submit that no new matter is being introduced into the application through the submission of this response.

Prior Art Rejections

The First 35 U.S.C. §103(a) Rejection

Each of claims 1-3, 5-8, and 10 was rejected under 35 U.S.C. §103(a) as being allegedly unpatentable over Kataoka (U.S. Pub. No. 2001/0056462) in view of Sun et al. (U.S. Pat. No. 6,442,663). As mentioned above, claims 7, 8 and 10 are being canceled; therefore, the rejection of claims 7, 8, and 10 is moot. Applicants respectfully traverse the rejection of claims 1-3 and 5-6 for the reasons set forth below.

Claim 1

Claim 1 is directed to a method for executing a job, which is loaded into a client machine, on a logical computer of a server machine by converting with the server machine environment information and job execution statements on the client machine side into the corresponding environment information and job execution statements on the server machine side.

In a method as recited in claim 1, the client machine is allowed to issue to the server machine a job execution request for executing the job. The job execution request is accompanied by environment information on a client machine side and job execution statements for the job to be executed. The environment information includes a volume

logical path, a volume physical path, a program product name, and a version of the program product on the client machine side. Furthermore, in a method as recited in claim 1, the server machine is allowed to allocate a logical computer for the job execution request, acquire both a volume logical path and a volume physical path on a server machine side for the allocated logical computer from the logical computer, assign a server side volume for the logical computer corresponding to a client side volume through a process of creating volume correlation information with use of the volume logical path and the volume physical path on the client machine side included in the environment information and the acquired volume logical path and the acquired volume physical path for the logical computer, control transfer of input data on the client side volume to a server side volume based on the volume correlation information, convert the environment information and the job execution statements based on the volume correlation information so as to replace information about the volume logical path and the volume physical path included in the job execution request by corresponding information for the logical computer of the server machine where the job is to be executed, and further replace the program product name and the version by corresponding information for the server machine on an as-needed basis, transmit the replaced environment information to said logical computer, and execute the job in said logical computer using the input data and the replaced environment information. As such, the job is executed by the server machine, NOT the client machine.

In contrast, Kataoka is directed to a system in which a client machine executes an application program loaded from a server. Kataoka shows that in response to an application program starting request made on a client machine, the server sends the client an execution environment identifying applet, a code which the application program to be started on the client machine requires, and a starting command for starting the application program. The client machine then executes the starting command on the client machine and starts the application program on the client machine. (See Fig. 2, paragraphs [0012]-[0014], and claim 1.)

Kataoka further shows that the client machine, into which a job is loaded, executes the job in the machine using the execution environment and the application program code both loaded from the server. In Kataoka, the client machine does not convert the execution environment information involved in the application program into the corresponding execution environment information on the client side, because the client environment is constructed on the basis of a Java virtual machine. Therefore, as admitted by the Examiner,

Kataoka cannot and does not show transfer of the input data from a client site volume to a server side volume based on the volume correlation information and conversion of the job execution statements and the environment information based on the volume correlation information.

The secondary reference of Sun is directed to a process migration from a first computer to a second computer. Sun shows a) collecting data associated with a process, which is in execution on the first computer, b) generating a memory representation data structure correlating logical parameters of the process to physical memory locations in the first computer, c) generating a type information table correlating data types associated with the logical parameters of the process to physical memory locations in the first computer, d) identifying which logical parameters will be needed to continue execution of the process on the second computer, e) saving the values and types of the identified logical parameters of the process in machine-independent format, and f) restoring the data in the identified physical memory locations of the second computer based on the values and types of the identified logical parameters of the process in machine-independent format and the correlations represented by the type information table. (See col. 29, line 64 – col. 30, line 27, col. 31, lines 5-9.)

The Examiner apparently asserted that the volume logical path and the volume physical path recited in claim 1 may read on logical parameters of the process and physical memory locations shown in Sun, respectively. Applicants respectfully disagree. A volume logical path and a volume physical path recited in claim 1 are paths for accessing to a disk volume. In contrast, although Sun does not explicitly state the definition of “the logical parameters” of the process, the logical parameters shown in Sun appears to be a logically viewed snapshot of a process memory space wherein each node in the snapshot represents a memory block allocated for storing data during program execution. (See col. 9, lines 1-25, col. 30, lines 11-14, and Fig. 3.) Thus, the logical parameters shown in Sun are clearly irrelevant to an access path to a disk volume. Also, physical memory locations shown in Sun appear to be physical memory addresses that are machine-dependent. Thus, Sun’s physical memory locations are irrelevant to a volume physical path as recited in claim 1.

Furthermore, in Sun, the values and types of the logical parameters which will be needed to continue execution of the process on the second computer is saved in machine independent format, and then the data in the physical memory locations of the second computer is restored in the physical memory locations of the second computer, based on the

saved values and types of logical parameters and correlation in the type information table. Namely, Sun shows the conversion of the data in the memory, but NOT conversion of a disk volume. Indeed, Sun is directed to process migration for checkpointing (See col. 2, lines 41-47.). It is apparent that Sun is not directed to job migration, because to perform job migration, the job execution environment has to be prepared before creating a process to execute the job; Sun does not do so.

In sum, the secondary reference of Sun fails to provide any disclosure, teaching or suggestion that makes up for the deficiencies in Kataoka. Therefore, at the time the invention was made, one of ordinary skill in the art could not and would not achieve all the features as recited in claim 1 by combining Sun with Kataoka. Accordingly, claim 1 is not obvious in view of all the prior art cited.

Claims 2-3

Claims 2-3 have substantially the same features as those of claim 1, at least with respect to features of the server machine. As such, the arguments set forth above are equally applicable here. Particularly, a client machine as recited in claim 2 further determines whether the job is to be executed on the client machine or the server machine according to policy information. Neither Kataoka nor Sun shows or suggests this feature.

Accordingly, claim 1 being allowable, claims 2-3 must also be allowable.

Claims 5-6

As to dependent claims 5-6, the arguments set forth above with respect to independent claim 3 are equally applicable here. The corresponding base claim being allowable, claims 5-6 must also be allowable.

The Second 35 U.S.C. §103(a) Rejection

Each of claims 4 and 9 was rejected under 35 U.S.C. §103(a) as being allegedly unpatentable over Kataoka in view of Sun, and further in view of Loomans (U.S. Pat. No. 6,393,605). As mentioned above, claim 9 is being canceled, and therefore, the rejection of claim 9 is moot. Applicants respectfully traverse the rejection of claim 4 for the reasons set forth below.

As set forth above, the combination of Kataoka and Sun fails to teach all the elements recited in claim 3, from which claim 4 depends. The secondary reference of Loomans fails to

provide any disclosure, teaching or suggestion that makes up for the deficiencies in the combination of Kataoka and Sun. Therefore, at the time the invention was made, one of ordinary skill in the art would not and could not achieve all the features as recited in claim 3, from which claim 4 depends. Accordingly, claim 4 is not obvious in view of all the prior art cited.

Conclusion

In light of the Amendments and Remarks, Applicants respectfully request early and favorable action with regard to the present application, and a Notice of Allowance for all pending claims is earnestly solicited.

Favorable reconsideration of this application as amended is respectfully solicited. Should there be any outstanding issues requiring discussion that would further the prosecution and allowance of the above-captioned application, the Examiner is invited to contact the Applicants' undersigned representative at the address and phone number indicated below.

Respectfully submitted,

Stanley P. Fisher
Registration Number 24,344



Juan Carlos A. Marquez
Registration Number 34,072

REED SMITH LLP
3110 Fairview Park Drive
Suite 1400
Falls Church, Virginia 22042
(703) 641-4200

August 11, 2008
SPF/JCM/YOM